

(19)日本国特許庁(JP)

(12) 公開特許公報(A)

(11)特許出願公開番号

特開平11-317735

(43)公開日 平成11年(1999)11月16日

(51)Int.Cl. <sup>8</sup>	識別記号	FI		
H04L 9/08		H04L 9/00	601F	
G06F 13/00	351	G06F 13/00	351Z	
G09C 1/00	640	G09C 1/00	640D	
H04L 9/32		H04L 9/00	673E	
			673B	

審査請求 未請求 請求項の数25 OL (全 17 頁) 最終頁に続く

(21)出願番号 特願平11-36453

(22)出願日 平成11年(1999)2月15日

(31)優先権主張番号 024928

(32)優先日 1998年2月17日

(33)優先権主張国 米国(US)

(71)出願人 599021000

シーサラムン ラマスブラマニ

SEETHARAMAN RAMASUB  
RAMANI

アメリカ合衆国, カリフォルニア州

95129 サン・ホゼ ミラー・アヴェニュー  
1195

(72)発明者 シーサラムン ラマスブラマニ

アメリカ合衆国, カリフォルニア州

95129 サン・ホゼ ミラー・アヴェニュー  
1195

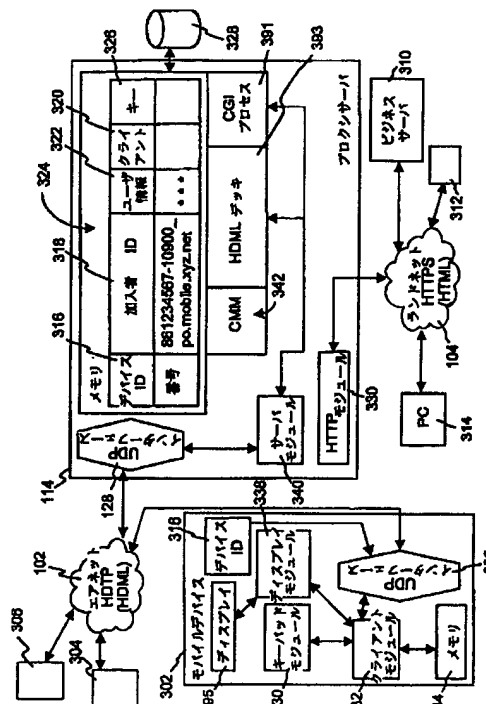
(74)代理人 弁理士 伊東 忠彦 (外1名)

(54)【発明の名称】 データネットワークにおける2方向インターラクティブコミュニケーションデバイスのための集中証明書管理システム

(57)【要約】

【課題】 データネットワークにおける貧弱なクライアントデバイスのための集中証明書管理システムを提供する。

【解決手段】 本発明は、サーバデバイスに貧弱なクライアントデバイスのためのデジタル証明書を管理させる証明書管理モジュールを有する。各貧弱なクライアントデバイスのために証明書を取得する遅延を最小にするために、その証明書管理モジュールは認証局により署名された固定数の自由証明書と各秘密キーを証明書データベースの中に保持し、証明書更新メッセージに応じて頻繁に自由証明書を更新する。貧弱なクライアントデバイスのためにユーザアカウントが生成される度に、証明書管理モジュールは1つ又はそれ以上の自由証明書を証明書データベースから取得し、取得した証明書を生成されたアカウントに関連付け、その間、証明書管理モジュールは、証明書データベースを埋めるために認証局とともに新しい自由証明書を作成する。



## 【特許請求の範囲】

【請求項1】 データネットワークを介してプロキシサーバと接続された複数の貧弱（thin）なクライアントデバイスのための、該プロキシサーバデバイスにおける集中証明書を管理するための方法であって、

該方法は前記プロキシサーバによりアクセス可能な自由証明書データベースを保持し、該自由証明書データベースは認証局（CA）によって発行された複数の自由証明書を有し、各自由証明書は対応する公開キー及び対応する秘密キーを有し、

該方法は前記プロキシサーバによりアクセス可能なユーザアカウントデータベースを保持し、該ユーザアカウントデータベースは複数のユーザアカウントを有し、前記各貧弱なクライアントデバイスは複数のユーザアカウントのうちの一つと関連付けられ、各ユーザアカウントは該ユーザアカウントに割り当てられたデバイスID、公開及び秘密キーのリスト、及び、該ユーザアカウントに割り当てられた証明書のリストを有し、

該方法は前記証明書データベースから取り出された少なくとも一つの証明書を前記ユーザアカウントデータベースにおける各ユーザアカウントに加えることを特徴とする方法。

【請求項2】 前記プロキシサーバにおける証明書データベースの保持は、

前記証明書データベースにおける自由証明書の数が下の閾値より低い場合に証明書要求を受け、

新たな証明書を生成し、該新たな証明書の生成は、

該新たな証明書のための識別名を生成し、

該新たな証明書のための新たな公開キー及び新たな秘密キーを生成し、

前記証明書要求を前記CAに送信し、該証明書要求は生成された該新たな公開キーを有し、

CAにより署名された新たな証明書を受信し、

前記自由証明書データベースに前記新たな証明書を置くことからなる請求項1に記載の方法。

【請求項3】 前記ユーザアカウントデータベースの保持は、

新たな貧弱なクライアントデバイスが活性化されると、前記自由証明書データベースから前記自由証明書のうちの一つを取得し、

新たなデバイスID及び新たな加入者IDに関連した新たなユーザアカウントを構築し、

取得した前記自由証明書、対応する秘密キー及び公開キーを前記新たなユーザアカウントと関連付けることからなる請求項1に記載の方法。

【請求項4】 証明書更新要求を受信すると前記自由証明書データベースにおける前記自由証明書を更新することを更に有する請求項1に記載の方法。

【請求項5】 前記証明書更新要求を受信すると前記自由証明書データベースにおける自由証明書を更新するこ

とは、

該証明書更新要求が証明書取り消しリストである場合に、前記自由証明書データベースから無効な証明書を取り除くことからなる請求項4に記載の方法。

【請求項6】 前記証明書更新要求を受信すると自由証明書データベースにおける自由証明書を更新することは、

前記証明書更新要求における挿入／削除クエリに応じて前記自由証明書データベースから証明書を削除することからなる請求項4に記載の方法。

【請求項7】 有効なデバイスIDを有する貧弱なクライアントから新たに設定されたユーザ名とパスワードを受信すると、該有効なデバイスIDと関連付けられた前記ユーザアカウントデータベースにおけるユーザアカウントを更新することを含む請求項1に記載の方法。

【請求項8】 前記ユーザアカウントデータベースにおける前記ユーザアカウントは、インターネットを介して前記プロキシサーバに接続されたコンピュータから前記新たに設定されたユーザ名とパスワードによりアクセス可能である請求項7に記載の方法。

【請求項9】 有効なユーザ名とパスワードは前記ユーザアカウントにアクセスするために供給されなければならない請求項8に記載の方法。

【請求項10】 データネットワーク上で、複数の貧弱なクライアントデバイスのために、プロキシサーバデバイスにおける集中証明書を管理するための装置であって、該装置は、

自由証明書を生成するための証明書管理モジュールと、上の閾値に達するまで、該証明書管理モジュールからの該自由証明書を格納するための、該証明書管理モジュールと接続された自由証明書データベースと、ユーザアカウントデータベースと、

前記自由証明書データベースにおける前記自由証明書のうちの一つを、新たに活性化された貧弱なクライアントデバイスと関連付けられた該ユーザアカウントデータベースにおける新たなユーザアカウントに関連付けるための証明書割り当てモジュールとを有し、

前記ユーザアカウントデータベースは前記プロキシサーバデバイスによりアクセス可能であり、該ユーザアカウントデータベースは複数のユーザアカウントを有し、各貧弱なクライアントデバイスは該ユーザアカウントのうちの一つと関係付けられ、該ユーザアカウントは該ユーザアカウントに割り当てられたデバイスID及び証明書リストを有することを特徴とする装置。

【請求項11】 前記証明書管理モジュールは、前記証明書割り当てモジュールと通信する証明書エンジンと、新たな証明書のための一意の名前を生成する名前生成器と、  
該新たな証明書のための秘密キー及び公開キーを生成す

るキーペア生成器と、

該新たな証明書のために認証局と交信する証明書要求モジュールとを有し、該証明書要求は前記公開キーと前記一意の名前を有する請求項10に記載の装置。

【請求項12】 前記名前生成器は、タイムスタンプと加入者IDとを結合させる識別名生成器を有する請求項11に記載の装置。

【請求項13】 前記証明書管理モジュールは、証明書更新要求を受信すると前記自由証明書データベースを更新する請求項12に記載の装置。

【請求項14】 前記証明書更新要求は証明書取り消しリストを有する請求項13に記載の装置。

【請求項15】 前記証明書更新要求は更に挿入／削除クエリーを有する請求項14に記載の装置。

【請求項16】 前記装置は、前記プロキシサーバデバイスと接続したコンピュータネットワークと、

該コンピュータネットワークに接続されたクライアントコンピュータとを更に有し、該クライアントコンピュータは前記ユーザアカウントデータベースにおけるユーザアカウントにアクセスすることが可能である請求項10に記載の装置。

【請求項17】 データネットワークを介してプロキシサーバと接続された複数の貧弱なクライアントデバイスのための、該プロキシサーバデバイスにおける集中証明書を管理する方法であって、

該方法は前記プロキシサーバによりアクセス可能なユーザアカウントデータベースを保持し、該ユーザアカウントデータベースは複数のユーザアカウントを有し、前記貧弱なクライアントデバイスは該ユーザアカウントのうちの一つと関連付けられ、各ユーザアカウントは、デバイスID、該ユーザアカウントに割り当てられた公開及び秘密キーのリスト、及び該ユーザアカウントに割り当てられた少なくとも一つの証明書を有し、

該方法は、第1の貧弱なクライアントデバイスと関連した第1のユーザアカウントに割り当てられた第1の証明書を使用して第1の貧弱なクライアントデバイスから、前記プロキシサーバデバイスに接続された安全なサーバにアクセスすることを特徴とする方法。

【請求項18】 前記プロキシサーバによりアクセス可能な自由証明書データベースを保持し、該自由証明書データベースは認証局(CA)により発行される複数の自由証明書を有し、各自由証明書は対応する公開キーと対応する秘密キーを有する請求項17に記載の方法。

【請求項19】 前記プロキシサーバにおける証明書データベースの保持は、

前記証明書データベースにおける自由証明書の数が下の閾値より低い場合に証明書要求を受け、新たな証明書を生成し、該新たな証明書の生成は、該新たな証明書のための識別名を生成し、

該新たな証明書のための新たな公開キー及び新たな秘密キーを生成し、

前記証明書要求をCAに送信し、該証明書要求は生成された該新たな公開キーを有し、

CAにより署名された新たな証明書を受信し、前記自由証明書データベースに前記新たな証明書を置くことからなる請求項18に記載の方法。

【請求項20】 前記ユーザアカウントデータベースの保持は、

10 新たな貧弱なクライアントデバイスが活性化されると、前記自由証明書データベースから前記自由証明書のうちの一つを取得し、

新たなデバイスID及び新たな加入者IDを有する新たなユーザアカウントを構築し、

取得した前記自由証明書、対応する秘密キー及び公開キーを前記新たなデバイスIDを有する前記新たなユーザアカウントと関連付けることからなる請求項17に記載の方法。

20 【請求項21】 証明書更新要求を受信すると前記自由証明書データベースにおける前記自由証明書を更新することを含む請求項18に記載の方法。

【請求項22】 前記証明書更新要求を受信すると自由証明書データベースにおける自由証明書を更新することは、

該証明書更新要求が証明書取り消しリストである場合に、前記自由証明書データベースから無効な証明書を取り除くことからなる請求項21に記載の方法。

30 【請求項23】 前記証明書更新要求を受信すると自由証明書データベースにおける自由証明書を更新することは、

前記証明書更新要求における挿入／削除クエリーに応じて前記自由証明書データベースから証明書を削除することからなる請求項21に記載の方法。

【請求項24】 有効なデバイスIDを有する貧弱なクライアントから新たに設定されたユーザ名とパスワードを受信すると、該有効なデバイスIDと関連付けられた前記ユーザアカウントデータベースにおけるユーザアカウントを更新することを含む請求項17に記載の方法。

40 【請求項25】 前記ユーザアカウントデータベースにおける前記ユーザアカウントは、インターネットを介して前記プロキシサーバに接続されたコンピュータからアクセスされる請求項17に記載の方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、データネットワークにおけるサーバコンピュータとクライアントコンピュータ間のデータセキュリティに関し、特に、データネットワーク上の2方向インタラクティブコミュニケーションデバイスのためのデジタル証明書を、プロキシサーバにおいて管理するシステムに関する。モバイルデバイ

ス、セルラ電話、地上線(landline)電話及びインターネット装置コントローラのような2方向インタラクティブコミュニケーションデバイスにおいては、一般的に、コンピューティングパワー、メモリ及びグラフィック表示能力等のコンピューティング資源が限られている。

#### 【0002】

【従来の技術】インターネットにおける急成長のトレンドは電子商取引である。電子商取引は、次のものをいっしょに取り入れて設計された統合概念である。すなわち、幅広い領域のビジネス支援サービス、商品・製品・カスタマイズされた製品・特注の品及びサービスのための取引支援、注文及び調達支援システム、決済支援システム、管理情報及び統計報告システムであり、これらは全てインターネットを介して実施される。しかしながら、インターネットは、相互接続された世界中のコンピュータ及び電子デバイスの、公共の開かれた国際ネットワークであるということは良く知られている。安全にデータを送受信する能力は、インターネット上で電子商取引を行う上で基本的な要求となっている。その開かれたネットワーク上でビジネス取引を行うために、企業又は組織は、不正から自分自身及び顧客を守るためのアイデンティティ及び信頼を確立する効率的で信頼性の高い方法を持たなければならない。同様に、顧客は、インターネットに発信するかもしれない個人情報が送信先の企業以外の誰にも読まれることがない、ということに確信を持つ必要がある。

【0003】2つのに認証された側の間における個人的コミュニケーション又はビジネス取引を保証するために進行している努力の一つは、その2つの身元を、インターネット上で伝送されるデジタル情報を暗号化し署名するために使用され得る電子キーのペアと結びつけるデジタル証明書を使用することである。デジタル証明書は、与えられたキーを使用する権利を持っているという誰かの主張を確かめることを可能とし、それにより、認証されたユーザになりすまそうとして他人が偽のキーを使用することを防止することに寄与する。暗号化と共に使用されることにより、デジタル証明書は、開かれたネットワークを介して取引に参加する全ての参加者の身元を確認することにより、より完全なセキュリティの解決策を提供する。

【0004】デジタル証明書を使用する現在の構成は、インターネット上で、クライアントコンピュータとサーバコンピュータの2つのコンピュータの間を結び付けることであり、このことは、両コンピュータが物理的にそれぞれ自身の証明書を持ち、証明書を保持するためのメモリスペースが必要であるということ意味する。それらの証明書のうちの1つが無効(期限切れ、取り消し、又は使用不可)になった場合、その無効の証明書を有するコンピュータは、証明書発行機関から新しい証明書を

取得し得る。

#### 【0005】

【発明が解決しようとする課題】しかしながら、その取得プロセスは一般的に時間が数分かかり、多大なコンピューティングパワーを要する。2つのコンピュータ間のコミュニケーションセッションが確立されると、その2つのコンピュータは、相手の証明書を調べることによって互いに認証しあう。認証が成功すると、セッションキーが生成され、その2つのコンピュータ間を交流する全ての情報を暗号化するためのそのセッションキーを使用して、コミュニケーションセッションが開始される。その認証プロセスはまた多大なコンピューティングパワーを要する。

【0006】クライアントコンピュータが、モバイルコンピューティングデバイス、セルラ電話、地上線電話又はインターネット装置コントローラのような小さな2方向コミュニケーションデバイスの場合、上記の構成は適用し難い。ポータビリティと移動性を増加させるために、そのような2方向コミュニケーションデバイスの大部分は、小さく、軽く、消費電力を小さく、そしてできるだけ安価に設計される。そのような設計は、しばしば貧弱な(thin)設計と考えられ、それによりコンピューティングパワーは非常に限られ、典型的には、それは典型的なデスクトップ又はポータブルコンピュータにおいて備えられているものの1%以下であり、そのメモリ容量は一般的に250キロバイト以下である。そのことは、その貧弱なクライアントデバイスは多数の証明書を格納するための余計なメモリ容量を持たず、保持している証明書の1つが無効になった場合にリアルタイムで新しい証明書を取得するために必要なコンピューティングパワーを持たないということを意味する。従って、効率的に証明書を管理する機構をその貧弱なクライアントに提供する大きな必要性がある。

#### 【0007】

【課題を解決するための手段】上記の目的を達成するために本発明は次のように構成される。本発明は、データネットワークを介してプロキシサーバと接続された複数の貧弱(thin)なクライアントデバイスのための、該プロキシサーバデバイスにおける集中証明書を管理するための方法であって、その方法は前記プロキシサーバによりアクセス可能な自由証明書データベースを保持し、該自由証明書データベースは認証局(CA)によって発行された複数の自由証明書を有し、各自由証明書は対応する公開キー及び対応する秘密キーを有し、また、その方法は前記プロキシサーバによりアクセス可能なユーザアカウントデータベースを保持し、該ユーザアカウントデータベースは複数のユーザアカウントを有し、前記各貧弱なクライアントデバイスは複数のユーザアカウントのうちの一つと関連付けられ、各ユーザアカウントは該ユーザアカウントに割り当てられたデバイスID、公開及

び秘密キーのリスト、及び、該ユーザアカウントに割り当てられた証明書のリストを有し、更にその方法は前記証明書データベースから取り出された少なくとも一つの証明書を前記ユーザアカウントデータベースにおける各ユーザアカウントに加えることからなる。

【0008】また、上記目的を達成するために次のような構成としてもよい。本発明は、データネットワーク上で、複数の貧弱なクライアントデバイスのために、プロキシサーバデバイスにおける集中証明書を管理するための装置であって、該装置は、自由証明書を生成するための証明書管理モジュールと、上の閾値に達するまで、該証明書管理モジュールからの該自由証明書を格納するための、該証明書管理モジュールと接続された自由証明書データベースと、ユーザアカウントデータベースと、前記自由証明書データベースにおける前記自由証明書のうちの一つを、新たに活性化された貧弱なクライアントデバイスと関連付けられた該ユーザアカウントデータベースにおける新たなユーザアカウントに関連付けるための証明書割り当てモジュールとを有し、前記ユーザアカウントデータベースは前記プロキシサーバデバイスによりアクセス可能であり、該ユーザアカウントデータベースは複数のユーザアカウントを有し、各貧弱なクライアントデバイスは該ユーザアカウントのうちのひとつと関係付けられ、該ユーザアカウントは該ユーザアカウントに割り当てられたデバイスID及び証明書リストを有する。

【0009】更に、上記目的を達成するために次のような構成としてもよい。本発明は、データネットワークを介してプロキシサーバと接続された複数の貧弱なクライアントデバイスのための、該プロキシサーバデバイスにおける集中証明書を管理する方法であって、その方法は前記プロキシサーバによりアクセス可能なユーザアカウントデータベースを保持し、該ユーザアカウントデータベースは複数のユーザアカウントを有し、前記貧弱なクライアントデバイスは該ユーザアカウントのうちのひとつと関連付けられ、各ユーザアカウントは、デバイスID、該ユーザアカウントに割り当てられた公開及び秘密キーのリスト、及び該ユーザアカウントに割り当てられた少なくとも一つの証明書を有し、また、その方法は第1の貧弱なクライアントデバイスと関連した第1のユーザアカウントに割り当てられた第1の証明書を使用して第1の貧弱なクライアントデバイスから、前記プロキシサーバデバイスに接続された安全なサーバにアクセスすることからなる。

【0010】

【発明の実施の形態】本発明のこれら及び他の特徴、側面、及び利点は、以下の説明、請求項、及び添付図面にに関してより良く理解される。

(表記及び述語) 下記の本発明の詳細説明において、多くの具体的詳細は、本発明を完全に理解するために述べられる。しかし、本発明がこれらの具体的詳細なしに実

施され得ることは当業者にとって明らかになる。次に、良く知られた方法、手順、コンポーネント、及び回路は、本発明の側面をあいまいにすることを回避するために、詳細には説明されない。

【0011】下記の本発明の詳細な説明は、ほとんど、手順、ステップ、論理ブロック、処理、及び、ネットワークに接続したデータ処理デバイスに似ている他の記号表現に関して行なわれる。これらの処理の説明及び表現は、当業者が他の当業者にその仕事の内容を最も効率的に伝達するために使用する手段である。本発明は、データネットワークにおける2方向インタラクティブコミュニケーションデバイスのための集中証明書管理システムである。以下で詳細に説明される方法は、所望の結果に到達するプロセス又はステップの自己一貫したシーケンスである。これらのステップ又はプロセスは、物理量の物理操作を必要とするものである。通常、必要ではないが、これらの量は、コンピュータシステム又は電気的コンピューティングデバイスにおいて格納され、送信され、結合され、比較され、表示され、その他の操作がされ得る電気信号の形をとり得る。これらの信号を、ビット、値、エレメント、シンボル、オペレーション、メッセージ、項目、数等と称することは、主に一般的に使用されるという理由で、ときどき便利であることがわかる。これらの同様な用語の全ては適切な物理量と関連し、これらの量に適用される単なる便利なラベルである、ということは覚えておくべきである。下記の説明から明らかなように、他に特別に述べられていなければ、本発明を通して、“処理”又は“コンピューティング”又は“確認すること”又は“表示すること”等のような用語を使用した議論は、コンピューティングデバイスのレジスタ及びメモリの中の物理量として表されるデータを、同様にコンピューティングデバイス又は他の電気的デバイスの中の物理量として表される他のデータへ、操作、変換する、コンピューティングデバイスの動作及び処理を意味しているということが認識される。

【0012】(デジタル証明書の紹介) 時々、デジタルID又はセキュリティ証明書と称されるデジタル証明書又は証明書は、しばしばテキストファイルとして格納され、開かれたデータネットワーク上で2つの当事者間に安全なコネクションを確立するためのセキュアソケットレイヤ(SSL)プロトコルによって使用される情報である。最も単純な形において、証明書は公開鍵と名前を含んでいる。良く使用されるものとして、証明書はまた、期限の日付、証明書を発行した認証局の名前、シリアル番号、及びおそらく他の情報を含む。最も重要なことは、証明書が証明書発行者のデジタル署名を含むということである。デジタル署名とは、証明書の内容を確かめるために使用され得る暗号化された“指紋”である。

【0013】デジタル証明書、もしくは単に証明書は、認証局(CA)によって発行され、CAの秘密キーを用

いて署名される。デジタルIDとして最も広く受け入れられているフォーマットは、国際標準CCITT X.509により定義されている。従って、証明書は、CCITT X.509に準拠したアプリケーションにより読み書きされる。デジタル証明書は、公開キーと秘密キーという関連する2つのキーのペアに基く公開鍵暗号技術を使用している。公開鍵暗号においては、公開キーは、そのキーペアの保持者と通信を欲する者には誰にでも取得可能である。公開キーは、秘密キーにより署名されたメッセージを確認するか、又は、秘密キーを使うことによってのみ復号化され得るメッセージを暗号化するために使用され得る。このようにして暗号化されたメッセージの安全性は、秘密キーの安全性による。秘密キーは不正使用から保護されなければならない。

【0014】証明書におけるキーのペアは、ユーザ名及び他の識別情報と結びつけられている。カリフォルニアのNetscape Communication Inc.のネットスケープナビゲータ、又は、ワシントンのMicrosoft CorporationのインターネットエクスプローラのようなHTMLブラウザにインストールされると、その証明書は、コンタクトされたサイトが調べることができる電子信任状として機能する。これは、時々、デジタル証明書が、会員権が必要な情報やサービス、又は、特定のユーザへのアクセスを制限するためのパスワードダイアログと置き換わることを可能とする。例えば、誰かがメッセージを業者のウェブサイトに送る場合、その人はそのメッセージに署名し、そのメッセージが確かにその人から送られたものであるということをそのメッセージの受取人に確信させるためにデジタルIDを含める。その業者がデジタル的に署名されたメッセージを受信すると、その署名者のデジタルIDは、偽造や不正な表現が起こっていないことを決定するために確かめられる。一般的に、一旦ユーザが証明書を取得すると、そのユーザは、その証明書を自動的に使用するために、セキュリティ機能付きウェブ又は電子メールアプリケーションを設定することができる。図1はクライアントと業者サーバ間でデジタルIDを使用する認証プロセスを示す。

【0015】認証の最も安全な使用法は、署名された各メッセージに1つかそれ以上の証明書を同封することを含む。そのメッセージの受信者は、認証局の公開キーを使用して証明書を確認し、送信者の公開キーであると確信し、そのメッセージの署名を確認する。メッセージには2つ又はそれ以上の証明書が同封されることがあり、階層的チェーンをなし、そこでは1つの証明書が前の証明書の確実性を保証する。証明書階層の終端はトップレベルの認証局であり、そこは他のいかなる認証局から認証なしで信頼される。トップレベル認証局の公開キーは、例えば、広く発行されることによって、独立して知られていなければならない。言い換えれば、受信者に会社名が知られている送信者は、ただ1つの証明書(そ

の会社により発行されたもの)を同封すればよいが、会社名が受信者に知られていない送信者は2つ又はそれ以上の証明書を同封する必要がある得る。セキュリティを高度にするために、チェーンにおける最も高いレベルの発行者が受信者に良く知られるのにちょうど十分の証明書チェーンを同封することは一般的な方法である。多数の受信者がいる場合、各受信者が必要とするものをカバーするために十分な証明書を含めねばならない。

(好ましい実施態様) ここで、図面を参照するにあたり、図面を通して同一の数字は同一の部分を示している。図2は、本発明の実施され得るデータネットワーク100を示す。データネットワーク100は、一般的にはワイヤレスネットワークと称されるエアネット(airnet)102、一般的には地上線(landline)ネットワークであるランドネット(landnet)104からなり、各々はそこを通るデータ伝送のためのコミュニケーション媒体である。エアネット(airnet)102では空中を介してデータ伝送が行われ、エアネット102はAT&T又はGTEのようなキャリアによって制御、運営されているので、ときどきキャリアネットワークと称される。各キャリアは、エアネット102において、CDPD、CDMA、GSM及びTDMAのような独自の通信スキームを有し得る。ランドネット104又はインターネットは、これらはここでは交互に用いられ、インターネット、イントラネット、又は他のプライベートネットワークであり得る。106により参照されるものは、1つのモバイルデバイス、セルラ電話、地上線電話若しくはインターネット装置コントローラであり得るモバイルデバイスのうちの1つであり、アンテナ108を介してエアネット102と通信できる。エアネット102は複数の2方向通信デバイスの通信を同時に運ぶが、ただ1つのモバイルデバイス106のみが図に示されている、ということは一般的に理解されることである。

【0016】同様に、複数のデスクトップパーソナルコンピュータ(PCs)110及び複数のサーバコンピュータ112がインターネット104に接続されているが、それぞれただ1つの例のみが図に示されている。図に示すように、PC110は、NEC Technologies Inc.のパーソナルコンピュータSPL300であり得、ハイパーテキストマークアップ言語(HTML)Webブラウザが載っており、そのブラウザは、インターネット104を介し、ハイパーテキストトランスファープロトコル(HTTP)を使用して、Sun Microsystems Inc.のワークステーションであり得るwebサーバに格納された情報にアクセスする。PC110はwebサーバにもなるためのアクセス可能な情報を格納し得るということは当業者には理解される。インターネット104及びエアネット102の間に、その間でデータを通信するプロキシサーバコンピュータ114がある。プロキシサーバコンピュータ114は、また、リンクサ

サーバ又はゲートウェイサーバコンピュータと称され、ワークステーション又はパーソナルコンピュータであり得、マッピング又は翻訳の機能を実行する。例えば、1つのプロトコルから他への通信プロトコルマッピングを実行し得、それゆえ、モバイルデバイス106は、サーバ112又はPC110のそれぞれのうちの1つと通信できる。

【0017】インターネット104で使用する1つのよく知られた通信プロトコルは、ハイパーテキストトランスポートプロトコル (HTTP) 又はHTTPのセキュアバージョンであるHTTPSであり、TCP上で動作し、サーバ114における、良く知られたハイパーテキストマークアップ言語ウェブブラウザもしくはHTMLウェブブラウザのウェブサーバ112への接続を制御し、その間での情報の交換を制御する。HTTPSは、HTMLブラウザとウェブサーバ間の安全で認証された通信において最も使用されるSSLをサポートしている。HTMLブラウザにおいて良く用いられる表示は、ユニバーサルリソースロケータ若しくはURLの前に“https”を使用するものであり、これはSSLコネクションが確立されることを示している。SSLコネクションにおける一方の側、好ましくはサーバ側はその相手側によって認証されなければならない証明書を有する。そして、各側は、それ自身、他の側、又は両側の証明書からの情報を使用して送信するものを暗号化し、目的とする受信者のみがそれを復号化でき、データが確かにそれが示す場所から来たということを他の側が確信でき、そのメッセージが改竄されていない、ということを確認める。

【0018】エアネット102を介したモバイルデバイス106とプロキシサーバ114の間の通信プロトコルはハンドヘルドデバイストランスポートプロトコル (HDTP) 又はセキュアアップリンクゲートウェイプロトコル (SUGP) であり、好ましくは、ユーザデータグラムプロトコル (UDP) 上で動作し、モバイルデバイス106においてHDMLウェブブラウザのプロキシサーバ114への接続を制御する。ここで、HDMLはハンドヘルドデバイスマークアップランゲージを意味する。HDMLは、HTMLと同様、タグベースのドキュメント言語であり、モバイルデバイス106の小さなスクリーンに情報がどのように表示されるかを定めるカード (card) において指定されるコマンド又はステートメントのセットからなる。通常、多数のカードがデッキ (deck) にまとめられ、そのデッキが、モバイルデバイス106とプロキシサーバ114の間で交換され得るHDML情報の最小ユニットである。”HDTP Specification”のタイトルのHDTPの仕様及び”HDML 2.0 Language Reference”のタイトルのHDMLは同封され、全体に参照されている。HDTPは、HTTPに似たセッションレベルのプロトコルである

が、そのオーバーヘッドを負わず、非常にコンピューティングパワーとメモリの限られた貧弱なデバイスにおける使用に高度に最適化されている。更に、UDPは、TCPの場合のように情報が交換され得る前にクライアントとサーバ間に確立されるコネクションを要しない。それゆえ、UDPを使用することは、クライアントとサーバ間でのセッション確立の間、多数のパケットを交換させる必要性を削減させる。トランザクションの間に交換するパケットが非常に少ないことは、非常に限られたコンピューティングパワーとメモリしか持たないモバイルデバイスが効率的にランドラインデバイスとやりとりするために望ましい特徴である。

【0019】モバイルデバイス106はディスプレイスクリーン116とキーボードパッド118から構成される。モバイル電話106における、マイクロコントローラ、ROM及びRAMを含むハードウェアコンポーネントは当業者には公知であるので、ハードウェアコンポーネントの詳細はここでは説明しない。スクリーン116とキーボード118を使用して、モバイル電話106のユーザは、エアネット102上をプロキシサーバ114とインタラクティブに通信できる。1つの実施形態によれば、コンパイルされリンクされた本発明のプロセスは、クライアントモジュールとしてROMに格納され、モバイルデバイス106をプロキシサーバ114と動作させる。キーボード118を使用した所定のキーシーケンスによる活性化により、マイクロコントローラは、ROMの中のクライアントモジュールを使用して、プロキシサーバ114への通信セッション要求を初期化する。通信セッションを確立すると、モバイルデバイス106は、典型的には、プロキシサーバ114から1つのHDMLデッキを受信し、RAMにキャッシュとしてそのデッキを格納する。上述した通り、HDMLデッキは1つ又はそれ以上のカードからなり、各カードは、ディスプレイスクリーン116上にスクリーンディスプレイを生成するために要求される情報を含む。カードデッキの中のカードの数は、モバイルデバイス及びエアネットネットワーク102におけるリソースの効率的な使用を促進するために選択される。一般的に、カードのうちの1つは選択カードであり、頻繁に訪れるウェブサイトのシーケンスを示し、ユーザに1つを選択させ、安全で認証された通信セッションがプロキシサーバとで確立する。このような通信セッションを確立させるために証明書を使用するプロセスは以下で説明される。

【0020】図3には、データネットワークにおける他の部分又はコンポーネントと相互に作用する本発明が示されている。エアネット102に接続される複数のモバイルデバイスの3つの表現が302、304及び306により参照され、同様に、ランドネット104に接続される複数の地上線デバイスの3つの表現が310、312及び314により参照される。図2におけるプロキシ

サーバ114であり得るプロキシサーバデバイス128は、エアネット102をランドネット104に接続する。従って、どのようなモバイルデバイスも、エアネット102、プロキシサーバ114及びランドネット104を介して地上線デバイスと通信できる。本発明の説明を容易にするために、モバイルデバイス302とリンクサーバ114の内部の構成がそれぞれ示されている。他のプロセス及びハードウェアは当業者には公知であるので、わかりやすさのためにそれらは詳細には示さない。

【0021】モバイルデバイス302のような各モバイルデバイスにはデバイスID316が割り当てられる。デバイスID316は電話番号や、IPアドレスとポート番号の組み合わせ、例えば、204.163.165.132:01905であり得る。ここで、204.163.165.132がIPアドレスであり、01905がポート番号である。デバイスID316は更に加入者ID318と関連し、加入者ID318は、プロキシサーバ114においてユーザアカウント324を確立させることによってモバイルデバイス302を活性化させる手順の一部としてプロキシサーバ114の中でキャリアにより認証される。加入者ID318は、例えば、AT&Tワイヤレスサービスによる861234567-10900\_pn.mobile.att.netの形をとり得る。加入者ID318はモバイルデバイス302の一意の識別子である。言い換えれば、各モバイルデバイス302、304及び306は、プロキシサーバ114における各ユーザアカウントを示す加入者IDに対応するそれぞれの一意のデバイスIDを有する。以下の説明は、モバイルデバイス302及び関連するアカウント324に基き、その説明はプロキシサーバ114と同時に通信する複数のモバイルデバイスに同様に適用されることは当業者にとって認識されるものである。

【0022】アカウント324は、デバイスID316又は加入者ID318により指し示され、URLのようなアドレス識別子により識別されるものであり、ユーザ情報322、証明書リスト320及び秘密キーリスト326からなるデータ構造である。ここで、ユーザ情報322はアカウント構成と、ユーザ名とパスワードのような他のアカウントに関連する情報を含む。アカウントのURLは、例えばwww.att.com/Pocketnetの形をとり得、これは、エアネット102がAT&Tワイヤレスサービスにより運用されていることを示している。証明書リスト320は、1つ又はそれ以上のCAにより発行された指定された証明書のリストを含むか又は指し示し、秘密キーリスト326はキーのリストを含み、それぞれのキーは証明書リスト320における各証明書に対応する。証明書リスト320における全ての証明書は、排他的に特定のアカウントと関連している。一般的に、プロキシサーバ114は多数のそのようなユーザアカウントをデータベース328に保持し、各ユーザアカウントは、同一のキャリアに加入しプロキシサーバ114によ

ってサービスを受けているそれぞれのモバイルデバイスと関連している。証明書はアカウントとそれぞれ関連しているため、1つのアカウントにおける証明書は他のアカウントにおける証明書と異なることは認識され得る。

【0023】CAから証明書を取得し、秘密キーと公開キーのキーの組を生成するためには、通常のフルパワーデスクトップコンピュータにおいて著しく長い時間がかかる。モバイルデバイスを使用して証明書を取得する時間の長さを最小にするために、(証明書管理モジュール(certificate manager module))CMM342が証明書データベースを、好ましくはデータベース328に保持し、1つ又は異なるCAからの、自由証明書と称される、指定されていないが発行された証明書のリストを保持する。証明書を必要とするウェブサーバにアクセスするために1つ又はそれ以上の証明書を必要とするモバイルデバイスを活性化するために、ユーザアカウントが作成される度に、証明書要求信号(certRequest)が、証明書データベースから必要な証明書を取得するために、CMM342に送られる。証明書データベースから取得された証明書を受信すると、CMM342は、デバイスID316と他のアカウント情報を付加することによって、証明書を特定のアカウントに割り当てる。それゆえ、取得された証明書は、特定のアカウントと結び付けられ、証明書リスト320に置かれる。その間に、CMMは証明書データベースの中で使用可能な自由証明書の数を調べ、その数が閾値と称されるある値以下、例えば200以下であれば、CMMはHTTPモジュール330を呼び、ランドネット104を介して適切なCAとコネクションを確立し、閾値に達するまで新しい自由証明書を取得して証明書データベースを満たす。そのようにして、証明書データベースには、新しいアカウントにすぐに使用できる自由証明書を救急するために、常に十分な自由証明書が使用可能となっている。

【0024】普通は十分なコンピューティングパワーを有するローカルデバイスにおいて証明書を取得する従来の方法とは異なり、更に、ローカルデバイスにユーザアカウントを物理的に格納する従来の方法と異なり、本発明は、証明書を非同期に取得するタスクを実行するためにプロキシサーバのコンピューティングパワーを使用しており、ユーザアカウントにおける証明書をプロキシサーバに保持している、ということは当業者にはここで認識され得る。全てのユーザのためにプロキシサーバにおいて証明書を管理することは、クライアントが、追加のコンピューティングパワー及びメモリを要求することなく、安全なウェブサイトにアクセスすることを可能する。他の利点は下記の説明において認識されるだろう。

【0025】証明書は、証明書及びそれと関連したものを与えられたキーを用いて発行する先のアイデンティティを保証する信頼された中央管理であり得るCAによって発行されるものである。例えば、会社又は大学はそれ



自身の従業員や学生に証明書を発行する。CMM342が自由証明書を取得する先のもの以外に、モバイルデバイス302が証明書をCAから取得する必要性に対応するために、サーバモジュール340は、ユーザに、ランドネット104に接続された、例えばPC314のようなコンピュータを介してモバイルデバイス302に関連するユーザアカウント324にログオンさせる。これは、例えばwww.att.com/Pocketnetのようなユーザアカウント324のアドレス識別子を使用してユーザアカウント324にログオンすることにより達成される。アカウント324が許可されたユーザによってアクセスされていることを確認するために、ユーザ名とパスワードのような信用情報の組が要求される。ユーザがhttp://www.att.com/Pocketnetを使用してPC314をURLに接続すると、HTTPモジュール330を介してサーバモジュール340はユーザ名とパスワードを促す。マッチするユーザ名とパスワードの組の入力はそのアカウントにアクセスするための許可として認められる。

【0026】アカウントに柔軟性とセキュリティを与えるために、ユーザ名とパスワードはユーザにより完全に管理される。モバイルデバイス302のユーザは、HTMLブラウザを備えたモバイルデバイス302を用いてプロキシサーバ114におけるデバイスアカウント324にアクセスすることができる。アカウントのURLを知れば、ユーザは、クライアントモジュール332にURLとデバイスID316からなる要求をUDPインターフェース336へ送信させるために、所定のキーを押す、UDPインターフェース336は次にHDTTPを使用してプロキシサーバ114への通信セッションを確立する。その要求は、プロキシサーバ114における対応するUDPインターフェース128により受信され、デバイスIDが許可されているかどうかを調べるためにサーバモジュール340により実行される。そして、プロキシサーバ114は、モバイルデバイス302に送信される返答を用いて、ユーザ名とパスワードの要求を確認する。その返答は、アカウントへのアクセスを許可するためのユーザ名とパスワードの組を、ユーザから要求するのではなく、実際は、アカウントへのアクセスの許可は、モバイルデバイス302からの要求におけるデバイスID316とプロキシサーバ114におけるアカウント320の格納されたデバイスIDとの一致により得られたのである。

【0027】そのかわり、その返答は、新しいユーザ名とパスワードの組を入力することによりユーザがアカウントを自分で設定することを許容する。アカウント320が新しいユーザ名とパスワードの組を受信すると、アカウント、すなわちユーザ情報322は更新される。自分での設定の手順の後、ユーザは、好ましくは十分なコンピューティングパワーを有し、より慣れたHTMLブラウザを備えているPC314を使用し、HTTP及び

アカウントへのURLを使用して通信セッションを確立し得る。新たに設定されたユーザ名とパスワードは、促されたときにPC314に入力され、HTTPを使用してプロキシサーバ114へパケットフォーマットで送られる。プロキシサーバ114では、HTTPサーバ330がユーザ名とパスワードを抽出し、サーバモジュール340は、メモリの中のユーザ情報322を用いて承認チェックを実行する。入力されたユーザ名とパスワードが一致すれば、承認され、ユーザ又はPC314はアカウント324にアクセスすることを許可される。ここで、ユーザは特定のCAから証明書を要求でき、証明書リスト320とキーリスト326を更新できる。HTMLブラウザを使用してCAから証明書を取得するプロセスは、当業者には公知であり、従って、ここでは説明されない。CMMにより提供される機能に加えて、自分で設定する機能により、モバイルデバイス302に指定された全ての証明書を保持するためにプロキシサーバを信頼しながら、ユーザがその必要に応じて証明書を作ることができる。

【0028】図3、図4及び図5には、モバイルデバイス302のユーザがユーザ特定のCAから証明書を要求する例が示されている。所定のキーが押された後、モバイルデバイス302はHDTTPを使用し、モバイルデバイス:30286123456-10900\_pn.mobile.xyz.netを示すアカウントのURLを使用してプロキシサーバ114に接続するための要求を行う。デバイスID86123456-10900はその要求から抽出され、同じデバイスID86123456-10900により示されるアカウント324があることを確認する。確認において、モバイルデバイス302のユーザはユーザ名とパスワードの組を促される。ユーザ名とパスワードはモバイルデバイス302がアカウント324にアクセスするために必要な情報でなく、ユーザがユーザ名とパスワードを管理する許可を得るものである、ということは説明した。ユーザが新たなユーザ名とパスワードを入力しなければ、ユーザアカウント324におけるユーザ名とパスワードは同一のままである。ユーザが新たなユーザ名とパスワード、例えばユーザ名”スミス”、パスワード”123456”の組を入力すると、アカウント324は新しいユーザ名とパスワードに更新される。ここで、ユーザはアカウント324を操作するために、ランドネット104の中のいかなるコンピュータにも行くことができる。PC314は、ユーザにより効率的にアカウント324を操作させるための全グラフィカルユーザインターフェースを提供するHTMLブラウザを備えている。PC314は、サーバモジュール340の中のゲートウェイ354のURL、例えばmobile.xyz.netを使用して、プロキシサーバ114における全てのユーザアカウントへのHTTPコネクションを確立する。ユーザは、ユーザ名とパスワードの組をPC314にて促される。ユーザは、ゲートウェイ354を通

過するために、ユーザ名として” スミス”、パスワードとして” 123456”を入力しなければならない。入力されたユーザ名とパスワードを受信すると、ゲートウェイ354はアカウント324におけるものと比較する。ミスマッチの場合、PC又はユーザはアカウント324へのアクセスを許可されない。入力されたユーザ名とパスワードがアカウント324におけるものと一致すれば、ゲートウェイ354はPC314へ許可を与える。PC314のユーザは、特定のCA358のURLを与えることによって、特定のCA358から特定の証明書<sub>10</sub>を要求するためにHTMLブラウザを使用して、使用するモバイルデバイス302のためアカウントに証明書を置くことができる。

【0029】本発明の一つの実施の形態によれば、証明書リストは証明書テーブル368へのポインタとして実装され得る。図5に示すように、ポインタを使用することによって証明書リストの柔軟な容量を提供できる、<sub>10</sub>ということは当業者にとって認識され得ることである。証明書インデックス370は、全ての証明書と、証明書インデックス370において特別に要求された証明書のためのURLと関連する対応するURLリスト372を格納する空間を提供する。あるCAから証明書を受け入れるいくつかのサービスウェブサイトがある。例えば、www.financial.comにより識別される金融のウェブサイトはCA S1により署名される証明書のみをとる。自分でアカウントを設定することによって、ユーザは、CA

S1からの証明書を特に要求でき、その証明書を証明書テーブル368に置くことができる。後の使用において、モバイルデバイス302はwww.financial.comへのコネクションを確立するための要求を送信する。www.financial.comからなる要求がプロキシサーバデバイス114にて受信されると、URLは対応する証明書、この場合はCA S1による証明書を取得するために使用される。CA S1による証明書をを用いて、モバイルデバイス302はwww.financial.comによって識別されるウェブへアクセスできる。一般的に、CMM342によって取得された証明書382は、多くのウェブサイト<sub>20</sub>で受け入れられる一般的なもののであり、特定のURLと関連はしていない。言い換えれば、証明書テーブル368は、ユーザによって特別に要求される、376、378及び380によって参照されるような、多くの特定の証明書と、CMM342によって自動的に取得される、382によって参照されるような、1つ又はそれ以上の一般的証明書とを有し得る。

【0030】図6はCMM342における種々のコンポーネントのブロック図である。上述したように、CMMは証明書データベースに固定数の自由証明書を保持し、証明書データベースにおける自由証明書の数が閾値より小さくなるとすぐに、CAからHTTPサーバ330を通して新たな証明書を取得し始める。CMM342にお

いて他のコンポーネントの動作を管理する証明書エンジンが402により参照されている。モバイルデバイス302が活性化されると、そのアカウントは1つ又はそれ以上の証明書をロードするように要求される。自由証明書が証明書データベースから取得された後、証明書データベースにおける使用可能な自由証明書の数が閾値異以下であることを発見すると、エンジン402は、識別名生成器404又はDN生成器に、生成される新しい証明書にための一意の式別名を生成させる。

【0031】識別名は、CCITTX、509標準における標準形式の名前である。識別名は1つ又はそれ以上の関連識別名からなり、各関連識別名は1つ又はそれ以上の属性値主張(attribute-value assertion)からなる。各属性値主張は、属性識別子と対応する値情報からなり、例えば、CountryName (国名) = US、Organization (組織) = XYZ, Inc 又はOrganizationUnit (組織ユニット) = XYZ Service Division である。識別名の使用は、X.500ディレクトリツリーにおける要素を識別するためであり、ここで、ディレクトリツリーは、インターネットのための”ホワイトページ”<sub>20</sub> 一人、コンピュータ、サービス、及び電子メールアドレスのディレクトリを実装するために使用されている。そのディレクトリは階層的に構成されている。すなわち、トップには国際組織と国があり、国は州や地方に分けられ、それは更に種々の方法で分けられる。関連識別名は、ディレクトリツリーにおける1つのノードから下位のノードへのパスである。全体の識別名は、ツリーのルートから特定の要素を表す終わりのノードへのパスをトラバースする。ディレクトリのゴールは、インターネットにおける全ての通信要素を一意に命名する基盤を提供することであり、それゆえ、識別名において”識別”される。

【0032】識別名生成器404により生成された識別名が結局ユーザ名と関連付けられていることを確かめるために、識別名プレフィックス生成器406が、その識別名のためのプレフィックスを生成する。そのプレフィックスは一般的にタイムスタンプと加入者IDとの結合であり、例えば、861765228-9であり、タイムスタンプがいつ証明書要求がなされたかを示し、加入者IDは、活性化されたときに、モバイルデバイスに割り当てられる。識別名プレフィックス生成器406からのプレフィックスを用い、識別名生成器404からの識別名は一意でなければならない。言い換えれば、証明書データベースにおける各自由証明書はそれ自身の名前を有し、全ての名前は識別されなければならない。

【0033】証明書エンジン402は、公開キーと秘密キーのペアを生成するためにキーペア生成器412、若しくはKP生成器を行使する。供給される基となる情報を用いて生成される公開キーに基いて秘密キーを生成するライブラリ機能のセットを使用することにより、それを行う。産業標準に従うために、キーペア生成器412

において使用されるライブラリのセットは、Marine Parkway, Suite 500, Redwood City, CA 94065 に住所を持つRSA Data Security, Inc. により供給される。生成されたキーは一般的に、1110101100001...00101 のような、バイナリ数のシーケンスの形をとり、それらを生成するソースを知ることなく重ならない。一意の秘密キー及び公開キーのペアを生成するために、ソースとしての乱数がライブラリのセットに応じて提供されなければならない。乱数を得る多くの方法があることは当業者には理解される。一般的に使用される方法のうちの一つは、ハードコード化され得るノイズのソースから、若しくは、ネットワークトラフィック情報からの一方方向ハッシュ関数を通して乱数を生成する方法である。一方方向は、一つの方向（進行方向）は反対の方向（逆方向）よりも非常に簡単に実行できることを意味し、これにより、公開キーから秘密キーを得ることができなくなる。そのようなハッシュ関数の一つの例は、それ自身の値をある回数掛け、続いてモジュロ操作を行う。

【0034】証明書エンジン402は、証明書データベースの中の証明書ののための新しいエントリを作成し、キーペアからの対応する秘密キーはその新たなエントリに格納され、その間、証明書エンジン402は、証明書署名要求若しくはCRSを生成するために、生成された識別名及びキーペア生成器412から得られた公開キーを使用する。CSRはCAから証明書を要求するための公開の標準フォーマットである。CSRは、とりわけ、CAによって証明される公開キーとその公開キーと関連する識別名を含む。CSRは、標準形式で証明書要求の中にパッケージされたデータのバイナリブロックであり、HTTPを使用してHTTPモジュール330を通してCAに送信される。

【0035】証明書要求を受信すると、CAは提供されたその情報を確認し、証明書を署名することにより他の情報とともにユーザの公開キーの正当性を証明する。そして、CAは証明書応答を発行する。そこには署名された証明書又はエラーが含まれ得る。証明書応答がエラーを含む場合、これは、要求された証明書が失敗したことを意味し、新たなプロセスが開始されなければならない。証明書応答がCAからかえってきた場合、証明書エンジン402は受信した証明書から識別名を抽出し、証明書データベースの中の対応するエントリを、証明書格納ライブラリ408を通して更新する。この時点で、そのエントリは、公開キーが埋め込まれた署名された証明書と対応する秘密キーを含む。それは、自由証明書として参照されてきたものである。

【0036】モバイルデバイス302が活性化されると、デバイスのための証明書を生成するための要求が出される。証明書エンジン402は証明書データベースから自由証明書を取得し、デバイスIDと関連付ける。その関連付けは、device\_cert\_map\_tbl と呼ばれる、

好ましくはプロキシサーバ114のRAMの中にある分離した一時的なテーブルにエントリを作成することにより実行される。

【0037】証明書格納(CS)ライブラリ408、又はCSライブラリは証明書データベースを管理するために使用され、時々、CAから証明書廃止リストを受信する。証明書廃止リスト(CRL)は、予定の期限切れ日前に廃止された証明書のリストである。なぜ証明書が廃止されなければならない、CRLに置かれるかの理由はいくつかある。例えば、証明書の中で特定されるキーは損なわれたかも知れず、証明書の中で特定されるユーザはもはやキーを使用する権限を持たないかもしれない。より詳細には、キーに関連するユーザ名が "XYZ会社の副社長スミス氏" であり、スミス氏がその会社を去った場合、その会社は、彼にそのキーでメッセージに署名させたくないだろう。従って、その会社はCRLを証明書に置く。署名を確認するとき、その署名者の証明書が廃止されていないかを確認するために関連するCRLをチェックすることができる。このチェックを実行するための時間に見合うだけの価値があるかどうかは、署名された文書の重要性に依る。CRLは、CAによって保持され、CAにより発行された廃止証明書についての情報を提供する。しかしながら、無効な証明書はいかなる場合も受け入れられないので、CRLは現在有効な証明書のみをリストしている。廃止された証明書がその元の期限日を過ぎた時、それはCRLから除かれる。CRLは分散して保持されるが、CRLの集中貯蔵所があり得、それは、多くの組織からの最新のCRLを含むネットワークサイトである。

【0038】証明書ライブラリ408はそのようなCRLを受信し、CMM342によって維持されている証明書がリストにあると、証明書エンジン402に処理を行うように通知する。CMM342は固定数の自由証明書を有する証明書データベースを維持する。CMM342が証明書データベースからの証明書をユーザアカウントに関連付けるとき、その関連付けられた証明書は有効でなければならない。これは、CSライブラリ408を通して最初にCRLを調べることによって保証される。証明書データベースから取得された証明書がとにかくCRL上にあれば、取得された証明書は捨てられ、次の証明書が証明書データベースから取得される。CRLを用いた取得された証明書の照合調査は、取得された証明書がアカウントに関係付けられる前に、常にCSライブラリ408の中で実行される。CRLを用いた照合調査は網羅的比較であり得ることは当業者には理解される。その照合調査にかかる時間又は計算量は、その全てがモバイルデバイス416と非同期でプロキシサーバ114内で実施されているので、CRLの長さにかかわらず、許容できる。

【0039】本発明の実施の形態によれば、付録のソー

スコードリストがCMMにおける動作を表している。main () 関数が、HTTPクライアントベースの要求のサービスをするために必要なスレッドを生成するInitializeという名前の関数により初期化されたTCertEngine オブジェクトとコールを作成する。また、証明書データベースの中の証明書をモニタするスレッドも作る。そのスレッドが作成されると、それは可能なリソースをモニタし、証明書を生成するためにTCertHttpProtoの中のGenerateCertを呼ぶ。このスレッドは、証明書プールのために、データベースの中の新たなエントリを生成するためにTDBCertPoolを使用する。

【0040】GenerateCert関数は、識別名生成器から新たな識別名を取得する。それはまた、キーペア生成器から新しい公開／秘密キーの組を取得する。GenerateCertはCSRを構築するためにこの情報を使用した。そして、それは、THttpCertRequestの中のSendCSR メソッドを使用して、HTTP上でCAへの要求を発行する。証明書応答がCAから返されると、TDBCertPoolを使用して自由プールの中のエントリを更新する。

【0041】自由証明書がユーザアカウントと関連付けられる必要があると、TcertCreateCallbackの中のHandleCreateCertメソッドが呼び出される。そして、そのメソッドはdevice\_\_cert\_\_map\_\_tblの中の新たなエントリを作成するためにTDBDeviceMapの中の関数を呼ぶ。そして、呼び出した側に応答を返す。再発行スレッド、TCertReissueThreadは証明書を再発行するためにTcertHttpProtoにおけるReissueCertを呼ぶ。それは、自由プールにおける証明書及びデバイスに関連する証明書を廃止するためにTDBCertPool上のメソッドをTDBDeviceMapへ呼び出す。

【0042】図7と図8は、本発明の集中証明書管理システムの動作フローチャートを示し、図3、図4、図5と合わせて理解されるべきである。本発明のコンパイルされリンクされたプロセスはプロキシサーバ502にロードされ、プロキシサーバ502に集中証明書管理をさせる。プロキシサーバは、一般的に十分なコンピューティングパワーとメモリを備えたサーバコンピュータ若しくはデバイスであり、そのデバイスには、そのデバイスに他のコンピューティングデバイスへのサービスをさせるアプリケーションが搭載されており、従って、そのアプリケーションは一般的にサーバと称され、そのデバイス自身はここではサーバデバイスと称される。本発明のコンピューティングデバイスは、モバイルデバイス、セルラ電話及びインターネット装置コントローラであり得る貧弱なデバイスである。

【0043】504において、CMM324は、好ましくはサーバデバイス502におけるローカルの格納ドライブに格納されている証明書データベースを、サーバデバイス502に維持させる。証明書データベースは、CAによって署名された、ユーザアカウント又は貧弱なク

ライアントとまだ関連付けられていない固定数の自由証明書を保持している。データベースにすぐに使用可能な自由証明書を維持することによって、その貧弱なクライアントは、著しい時間遅延なく、コンピューティングパワー及びメモリの追加なく、関連する証明書を取得できる。506において、使用可能な自由証明書の数が調べられる。その数が落ちると、新たな証明書を取得するステップが510にて開始する。証明書データベースにおける自由証明書の数は、時々、508における証明書更新により落ちる、ということは理解されるべきである。ユーザアカウントと関連付けされた証明書が常に有効であることを確認するために、CSライブラリ408は、CA又は良く使用される貯蔵サイトから受信された証明書の更新メッセージに従って、証明書データベースにおける自由証明書をコンスタントに更新する。その証明書のメッセージはCRL又は挿入／削除クエリから構成され得、CMM324にいくつかの自由証明書を捨てさせ、それにより自由証明書の数が減少する。いかなる場合でも、CMM324は、CAから新しい証明書を取得することによって証明書データベースの中の自由証明書のレベルを保とうとする。新しい証明書を取得するプロセスが開始すると、CMM324は最初に、510と512において、DNプレフィックス生成器406とDN生成器404を呼び出すことにより、新たな証明書のための識別名を得、そして、514において秘密キーと公開キーの組を生成するためのKP生成器412を呼ぶ。証明書要求が、516において、生成された識別名と公開キーからなるCSRを含ませるために形成される。518において、CMM342は、HTTPサーバ330を介してHTTPを使用してCAと通信する。証明書要求を受信すると、CAは、証明書に署名することにより他の情報とともに公開キーの有効性を認証し、CMM342に証明書応答を返し、署名された証明書が520で作成される。署名された証明書は、522において、自由証明書として証明書データベースに預けられる。論理的には、自由証明書の数は1づつ増やされ（インクリメントされ）、固定数又は閾値と比較される。インクリメントされた数がまだ閾値以下である場合、新たな証明書を取得するプロセスは510から、証明書データベースの中の自由証明書の数が閾値に達するまで繰り返される。

【0044】その間、CMM342は536にて複数のユーザアカウントを維持し、それらのアカウントは好ましくはそれぞれ1つの貧弱なクライアントに割り当てられる。各アカウントは、そのアカウントに排他的に関連した1つ又はそれ以上の証明書を有する。貧弱なクライアントが、サーバデバイス502によりサービスを受けるために活性化されると、新しいユーザアカウントが538にてそのために確立される。前述したように、ユーザアカウントはデバイスID、加入者ID、ユーザ情

報、証明書リスト、秘密キーリストからなり得る。デバイスIDは、サーバデバイス502がどの貧弱クライアントデバイスにサービスするかを認識することを助ける情報であり、貧弱情報が活性化すると入力される。ユーザ情報は、貧弱クライアントが必要とするアカウント構成とサービスに関する情報を含む。加入者ID、証明書リスト及び秘密キーリストは、証明書がそれに関連付けられる時に得られる。540において、証明書を取得するための要求が作成される。542にて要求を受信すると、CMM342は証明書データベースから有効な自由証明書を取得し、自由証明書をアカウントに関連付ける。

【0045】本発明は自分で設定する方法を含む。特に、ユーザは図8におけるステップ544に示されるように自分での設定を試み得る。第一に、その試みはアカウントにログオンすることによりユーザアカウントにアクセスすることである。ユーザがデバイスIDを有する貧弱なクライアントデバイスを使用してログインする場合、アクセスはすぐに確認される。ユーザがインターネットに接続したPCを使用してログインする場合、今度は、ユーザは現在のユーザ名とパスワードを入力しなければならない。アクセスを取得した後、ユーザは、ステップ572にてユーザ名及び/又はパスワードを変更し得る。ユーザはそして、578において、ユーザ特定のCAから証明書を要求するために、貧弱なクライアント又は他のコンピューティングデバイスからアカウントにアクセスできる。

【0046】サーバデバイス502において1つ又はそれ以上の証明書を有するアカウントを構築することによって貧弱なクライアントが活性化した後、その貧弱なクライアントは、安全で認証された通信セッションを、いくつかの安全なウェブサイトと秘密通信を行うために、確立することができる。550において、サーバデバイス502は、URLで識別されたウェブサイトと安全で認証された通信セッションを確立するために、貧弱なクライアントからセッション要求を受信する。サーバデバイス502が貧弱なデバイスを認識し、その結果そのような要求を確認するために、そのセッション要求は、貧弱なクライアントのデバイスIDからなる。552において、デバイスIDはセッション要求から抽出され、ユーザアカウントの中のデバイスIDと比較される。デバイスIDが一致すれば、貧弱なデバイスは認証され、更に、対応するアカウント毎に調べられる。544において、一致したアカウントにおける証明書が取得され、HTTPSを使用して所望のウェブサイトに送られるセッション要求に含められる。558において、貧弱なクライアントと接続したウェブサイト間の認証はそれぞれ相手の証明書を調べることによって実行される。各証明書が信頼されると、セッションキーがそこから結果として生じ、その貧弱なクライアントとウェブサイト間で交換

される情報を暗号化するために使用される。

【0047】本発明は、ある詳細の程度をもって、十分に詳細に説明された。本実施の形態の開示は例のみを介してなされ、ステップとともに部分のアレンジメント及び組み合わせにおける多大な変更はクレームされた本発明の精神とスコープから離れることなく行なわれ得る、ということは当業者には理解される。従って、本発明のスコープは、一実施形態の説明よりも添付のクレームにより定義される。

10 【0048】(マイクロフィッシュ付録の参照) 本発明の開示の一部である付録Aは、"データネットワークにおける2方向コミュニケーションデバイスのための集中証明書管理システム"という題名の、全部で184フレームを有する2シートからなるマイクロフィッシュ付録である。そのマイクロフィッシュ付録は、本発明のワイヤレスデータネットワークにおける2方向コミュニケーションデバイスのための集中証明書管理システムの一実施例のソースコードリストであり、上記においてより完全に説明されている。

20 【0049】この特許文書の開示の部分は、著作権保護の対象となり、付録A、B、及びCを含むがこれに限定されないマテリアルを含む。著作権保持者は、それが特許商標の特許ファイル又は記録にあれば、本特許文書又は特許開示のいかなる者によるファクシミリ複製にも反対理由を持たない。しかしながら、それ以外の場合、著作権保持者は全ての著作権を保持する。

【図面の簡単な説明】

【図1】クライアントデバイスと業者のサーバ間で証明書がどのように使用されるかの模範を示す図である。

30 【図2】本発明が実施され得る、エアネット及びランドネットからなるモバイルデータネットワークを示す図である。

【図3】データネットワークにおいて他の部分及びコンポーネントと対話する本発明を示す図である。

【図4】モバイルデバイスのユーザが、ユーザの特定するCAから証明書を要求する例を示す図である。

【図5】モバイルデバイスのユーザが、ユーザの特定するCAから証明書を要求する例を示す図である。

40 【図6】本発明の認証モジュールにおける種々のコンポーネントのブロック図である。

【図7】データネットワーク上で、貧弱なクライアントのためにサーバデバイスにおいて証明書を管理するためのプロセス及び手順を示す動作フローチャートである。

【図8】データネットワーク上で、貧弱なクライアントのためにサーバデバイスにおいて証明書を管理するためのプロセス及び手順を示す動作フローチャートである。

【符号の説明】

100 データネットワーク

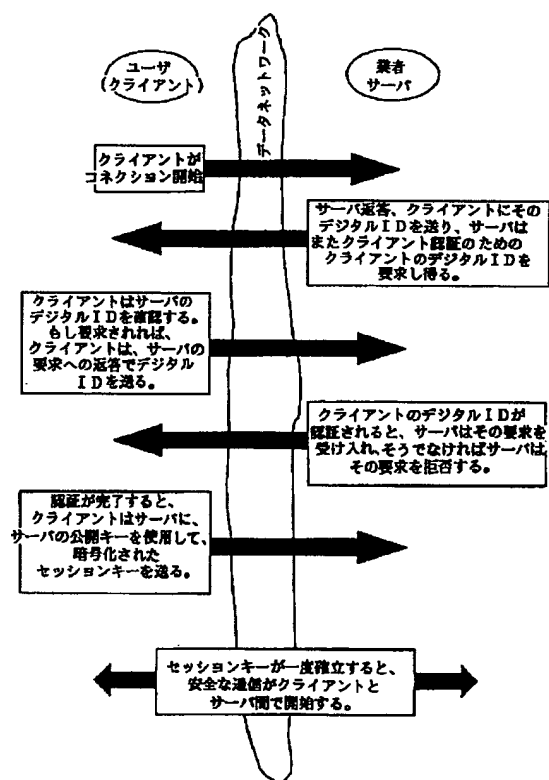
102 エアネット (airnet)

50 104 ランドネット (landnet)

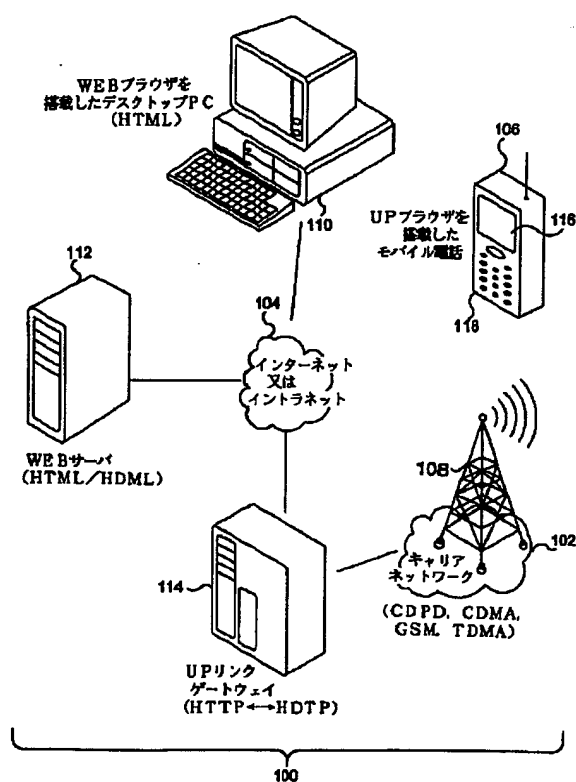
- 108 アンテナ
- 106、302、304、306 モバイルデバイス
- 110 デスクトップパーソナルコンピュータ
- 112 サーバコンピュータ
- 114 プロキシサーバコンピュータ
- 116 ディスプレイスクリーン
- 118 キーボードパッド
- 128 UDPインターフェース
- 310、312、314 地上線デバイス
- 316 デバイスID
- 318 加入者ID
- 320 証明書リスト
- 322 ユーザ情報
- 324 ユーザアカウント
- 326 秘密キーリスト
- 328 データベース328
- 330 HTTPモジュール
- 332 クライアントモジュール

- 3 3 4    メモリ
- 3 3 6    UDPインターフェース
- 3 4 0    サーバモジュール
- 3 4 2    CMM
- 3 5 4    ゲートウェイ
- 3 5 6、3 5 8    CA
- 3 6 8    証明書テーブル
- 3 7 0    証明書インデックス
- 3 7 2    URLリスト
- 10 3 7 4、3 7 6、3 7 8、3 8 0、3 8 2    証明書
- 4 0 2    証明書エンジン
- 4 0 4    識別名生成器
- 4 0 6    DNプレフィックス
- 4 0 8    証明書格納ライブラリ
- 4 1 0    証明書要求モジュール
- 4 1 2    キーペア生成器
- 4 1 4    シード生成器

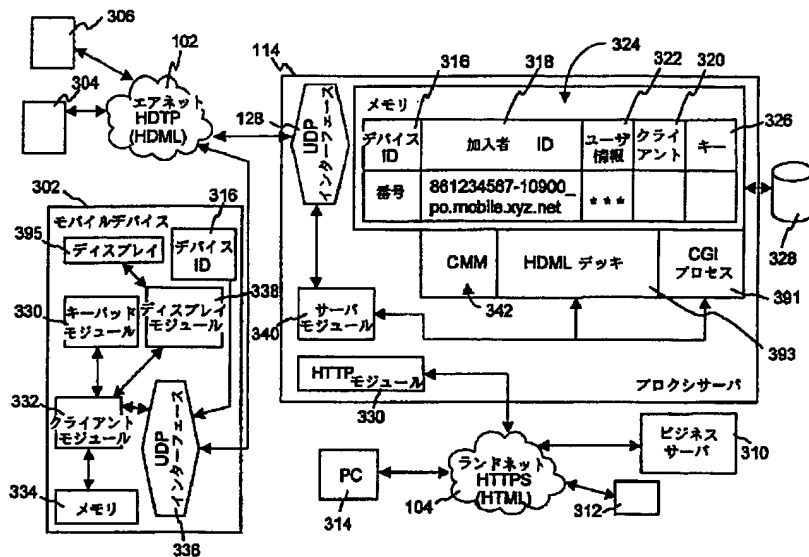
【图 1】



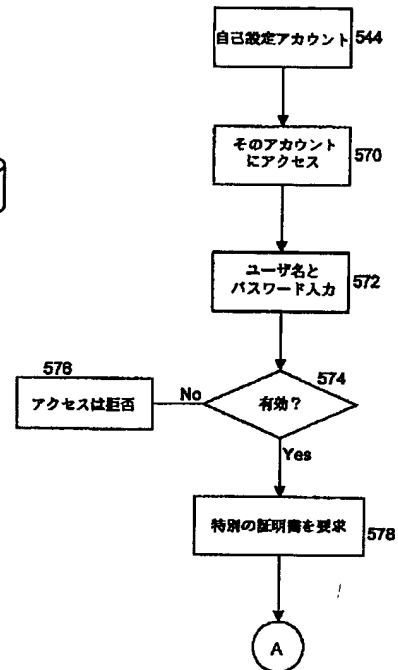
【图2】



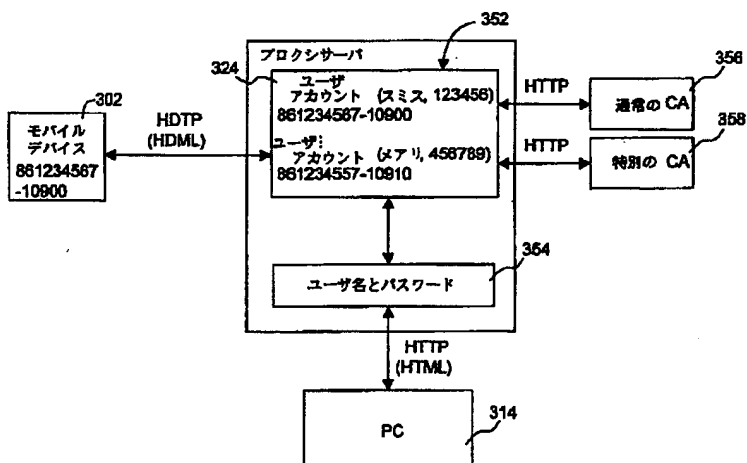
【図3】



【図8】



【図4】



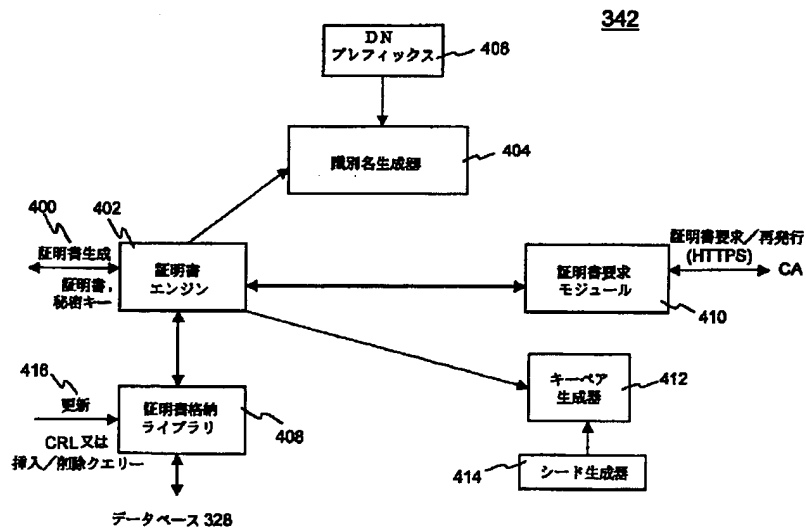
【図 5】

デバイス ID	加入者 ID	ユーザ情報 (smith 123456)	証明書 リスト	キー
	861234567-10900_pn.mobile.xyz.net			

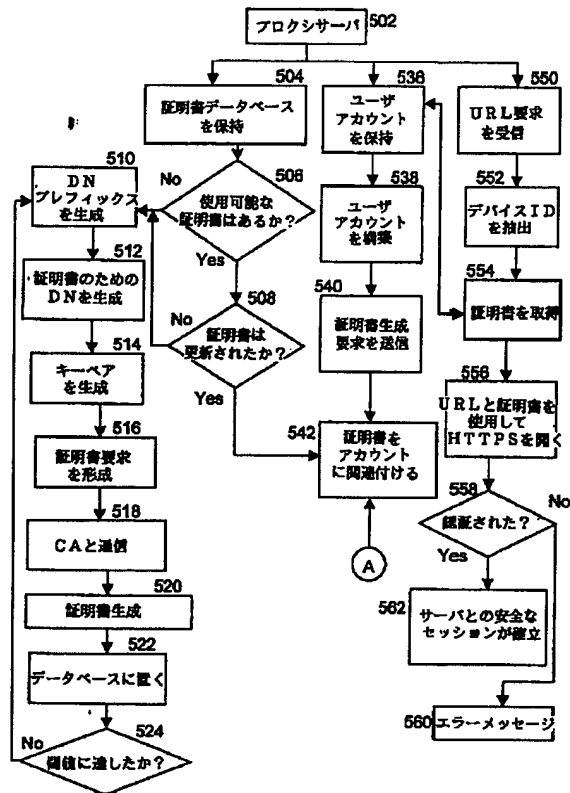
証明書	URL
CA S 1 による証明書	www.financial.com
CA S 2 による証明書	www.stock.com
CA S 3 による証明書	www.fund.com
CA S 4 による証明書	www.merchant.com
CA S 5 による証明書	

【図 6】





【図7】



フロントページの続き

(51) Int. Cl. 6

識別記号

F I

H 0 4 L 9/00

6 7 5 D